Part 1 Kalman Filter Tracking

A Kalman Filter is implemented to estimate and track the bounding box of a moving object in a video sequence The 8-dimensional state vector is defined as

$$x = \left[x, y, \omega, h, \dot{x}, \dot{y}, \dot{\omega}, \dot{h}\right]^T$$

, where (x, y) represents the top-left corner of the bounding box, (w, h) the width and height, and the corresponding derivatives are their velocities.

Prediction Step: The filter predicts the next state and error covariance as:

$$x_{k|k-1} = Fx_{k-1}, P_{k|k-1} = FP_{k-1}F^T + Q$$

Update Step: Upon receiving a measurement z_k, the update equations are:

$$K_k = P_{k|k-1}H^T(HP_{k|k-1}H^T + R)^{-1}$$

$$x_k = x_{k|k-1} + K_k(z_k - Hx_{k|k-1})$$

$$P_k = (I - K_kH)P_{k|k-1}$$

Kalman Filter is initialized with the first ground truth bounding box. The matrices used for initialization are:

- Initial state covariance: $P = diag([500,500,500,500,50,50,50,50])$
- Process noise: $Q = 0.1 \cdot I_s$
- Measurement noise: $R = diag([10,10,10,10])$

A small initial covariance P implies high confidence in the initial estimate, causing the Kalman Filter to rely heavily on its prediction model. This can slow adaptation and lead to divergence if the initial state is inaccurate. Using a larger P allows the filter to remain adaptive in the early frames and trust incoming measurements more, especially for velocity components which are initially unobserved.

The measurement model treats the observed bounding box as the measurement:

$$z_k = Hx_k + v_k \; with \; H = \begin{bmatrix} I_4 & 0_{4x4} \end{bmatrix}$$

The measurement noise covariance matrix R captures uncertainty in the measurement (bounding boxes). A smaller R indicates high confidence in the measurements, making the Kalman Filter correct predictions more aggressively. Conversely, a larger R implies lower confidence in the measurements, causing the filter to rely more on its internal predictions and adjust less.

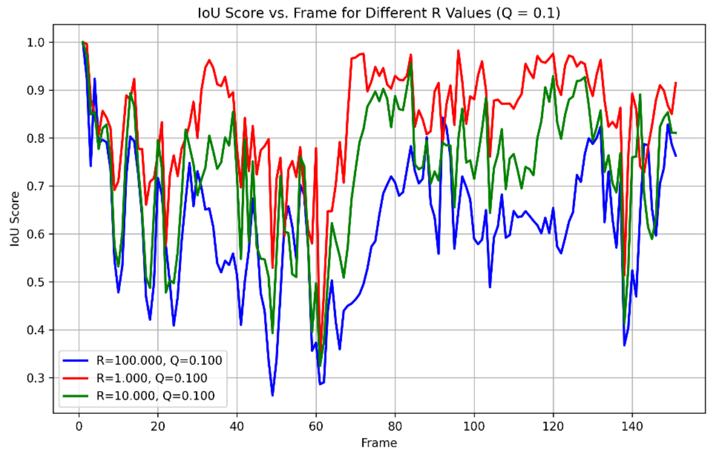

Figure 1 IoU Score vs. Frame for Different Measurement Noise Values (Q = 0.1) for butterfly sequence

In Figure 1, lower values of R result in better tracking performance, as reflected by higher IoU scores. The red curve (lowest R) demonstrates the most accurate tracking. However, extremely small R can lead to overreacting to noisy measurements, reducing stability. Thus, tuning R requires balancing responsiveness and robustness.

The process noise covariance matrix Q reflects uncertainty in the motion model. A small Q implies strong confidence in the constant velocity assumption, resulting in smooth but potentially lagging estimates. A large Q increases adaptability to abrupt motion changes but may introduce instability.

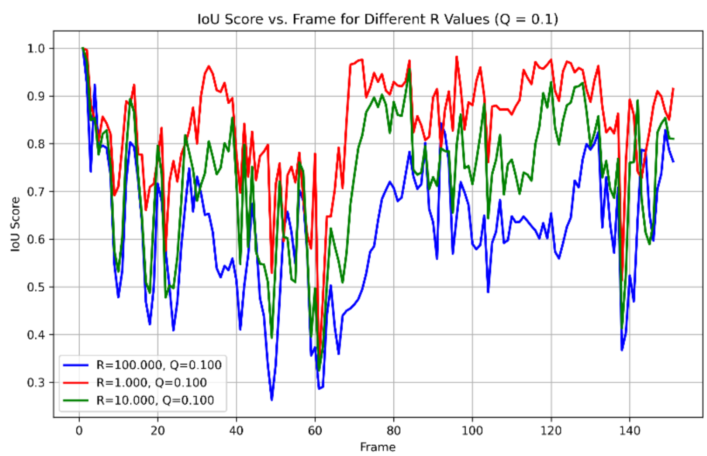

Figure 2 IoU Score vs. Frame for Different Measurement Process Noise Values (R = 10) for butterfly sequence

As shown in Figure 2, a moderate Q=0.1 provides the best performance. Too small a Q underestimates motion variability, leading to sluggish response. Too large a Q overestimates uncertainty, causing the filter to jitter or diverge. The plots for helicopter sequence is in Appendix.

Table 1 Kalman Filter Performance with Fine-Tuned Parameters (Q = 0.1, R = 10)

Dataset	**Avg IoU**	**Std. Dev**	**Min**	**Max**	**IoU ≥ 0.7**	**% Acceptable**
Helicopter	0.8273	0.1703	0.0000	1.0000	603 / 708	85.2%
Butterfly	0.7297	0.1370	0.3248	1.0000	103 / 151	68.2%

After evaluating the Kalman Filter under various process noise (Q) and measurement noise (R) configurations, the combination of Q = 0.1, R= 10 was found to yield the most robust and consistent tracking performance across both datasets. The sample frames and demonstration videos can be found in Appendix.

Table 2 Effect of Detection Frequency on Kalman Filter Tracking Performance

Sequence	**Interval**	**Avg IoU**	**Std. Deviation**	**IoU ≥ 0.7**	**% Acceptable**
Helicopter	1 (Dense)	0.8273	0.1703	603/708	85.2%
	5 (Moderate)	0.7118	0.2518	467 / 708	66.0%
	10 (Sparse)	0.6156	0.2920	342 / 708	48.3%
Butterfly	1 (Dense)	0.7297	0.1370	103/151	68.2%
	5 (Moderate)	0.5904	0.2076	53/151	35.1%
	10 (Sparse)	0.4430	0.2685	32/151	21.2%

For the detection frequency on tracking performance, Kalman Filter tracking was tested under different measurement intervals (1, 5, and 10). A smaller interval provides more frequent updates, while larger intervals simulate sparser detections. Results show that as the interval increases, average IoU decreases and standard deviation increases for both the helicopter and butterfly sequences. For example, in the butterfly frames, average IoU dropped from 0.73 (interval 1) to 0.44 (interval 10), and the percentage of acceptable frames (IoU ≥ 0.7) dropped from 68.2% to 21.2%. This trend highlights the importance of frequent measurements for maintaining accurate and stable tracking, especially in sequences with rapid or unpredictable motion.

Part 2: SiamMask Tracking

For deep learning-based tracking, the pre-trained SiamMask tracker was used. SiamMask is a one-shot visual object tracker that combines Siamese networks with instance segmentation. It was initialized using the first ground truth bounding box and applied to both the helicopter and butterfly datasets.

Performance-wise, SiamMask achieved an average IoU of 0.705 on the helicopter sequence and 0.666 on the butterfly sequence. These results demonstrate its effectiveness in scenarios where end-to-end learning-based tracking is preferred over model-based filtering approaches.

Table 3 Performance Comparison Between SiamMask and Kalman Filter on Helicopter and Butterfly Sequences

Method	**Sequence**	**Avg IoU**	**Std Dev**	**Min IoU**	**Max IoU**	**Frames with IoU ≥ 0.7 (%)**
SiamMask	Helicopter	0.7050	0.1768	0.0000	0.9915	383 / 707 (54.2%)
SiamMask	Butterfly	0.6664	0.1698	0.1300	0.9716	65 / 150 (43.3%)
Kalman Filter	Helicopter	0.8273	0.1703	0.0000	1.0000	603 / 708 (85.2%)
Kalman Filter	Butterfly	0.7297	0.1370	0.3248	1.0000	103 / 151 (68.2%)

Table 3 summarizes the tracking performance of both Kalman Filter and SiamMask across the helicopter and butterfly sequences. Overall, the Kalman Filter outperformed SiamMask in both average IoU and consistency. On the helicopter sequence, Kalman Filter achieved an average IoU of 0.8273 with 85.2% of frames exceeding 0.7 IoU, compared to SiamMask's 0.7050 and 54.2%.

Similarly, for the butterfly dataset, Kalman Filter reached 0.7297 average IoU versus SiamMask's 0.6664. These results indicate that, when well-tuned, the Kalman Filter can be highly effective, especially in scenarios with reliable detections.

However, SiamMask provides an end-to-end deep learning-based solution that is more robust to variations like object deformation and scale changes. It may perform better in challenging conditions or when trained on domain-specific data. In contrast, the Kalman Filter's performance is more sensitive to parameter tuning and assumes relatively smooth motion.

Table 4 Runtime Comparison Between Kalman Filter and SiamMask on the Butterfly Sequence

Method	**Sequence**	**Runtime (s)**	**FPS**
Kalman Filter	Butterfly	5.63	26.82
SiamMask	Buterfly	72.93	2.07

The Kalman Filter achieves real-time performance with around 27 FPS, while SiamMask, being a deep-learning based method with segmentation refinement, runs at around 2 FPS on the same sequence. This highlights the computational efficiency of Kalman Filters, making them suitable for real-time or resource-constrained applications, whereas SiamMask offers higher visual robustness at the cost of speed.

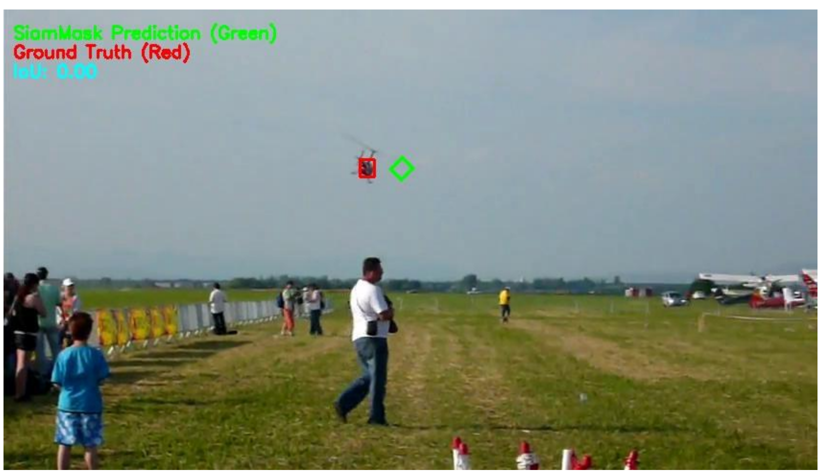

Figure 3 Tracking Failure and Recovery in SiamMask (Helicopter Sequence)

In challenging frames such as the one shown above, the helicopter becomes very small and hard to detect due to scale changes and motion blur. While the SiamMask tracker temporarily loses the target (resulting in low IoU), it is able to reacquire the object automatically in subsequent frames without any external intervention. This demonstrates SiamMask's robustness and ability to re-detect the object using visual cues learned from the initial template. In contrast, the Kalman Filter, which relies purely on motion prediction, continues to predict a bounding box, but without updated detections, it may drift or fail entirely during occlusions or erratic motion.

Appendix

Videos

- kf_butterfly
- siam_butterfly
- kf_helicopter
- siam_helicopter

Frames

- kf_butterfly_tracking_output
- kf_helicopter_tracking_output
- siam_butterfly_tracking_output
- siam_helicopter_tracking_output

Fine Tuning for the helicopter sequence

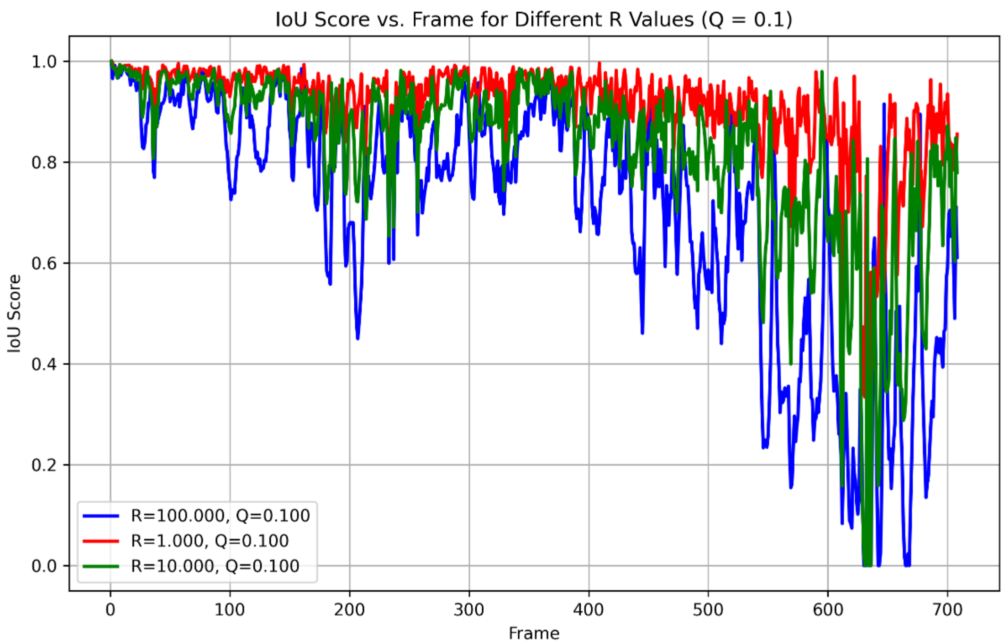

Figure 4 IoU vs. Frame (Different R values, Q=0.1) with Helicopter sequence using Kalman Filter

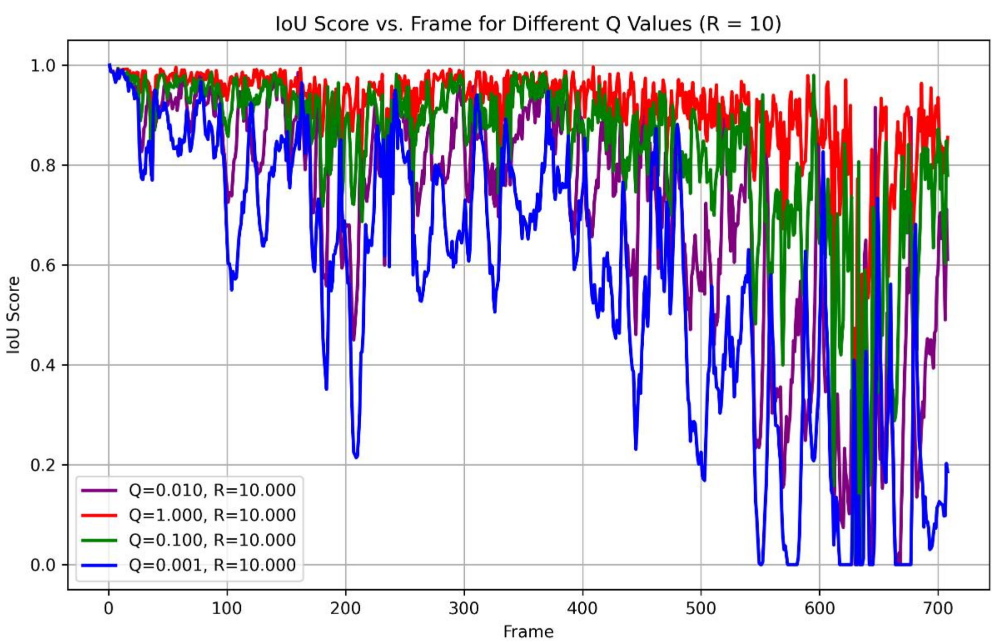

Figure 5 IoU vs. Frame (Different Q values, R=10) with Helicopter sequence using Kalman Filter

IoU plots of SiamMask

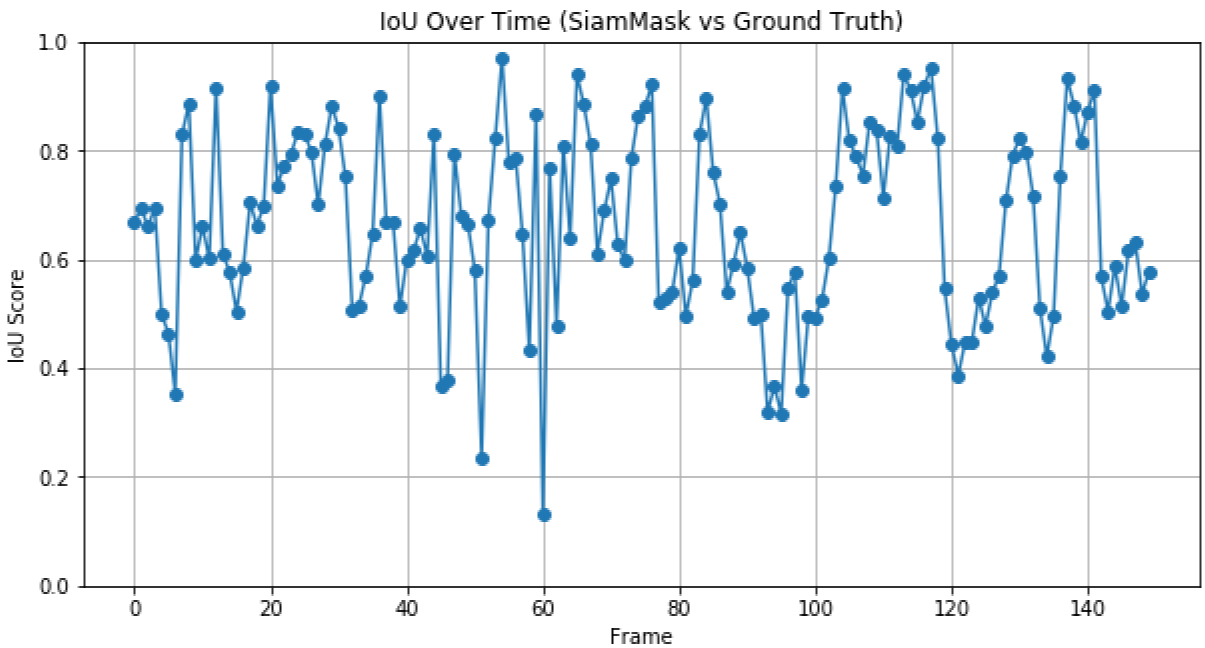

Figure 6 IoU vs. Frame with butterfly sequence using SiamMask

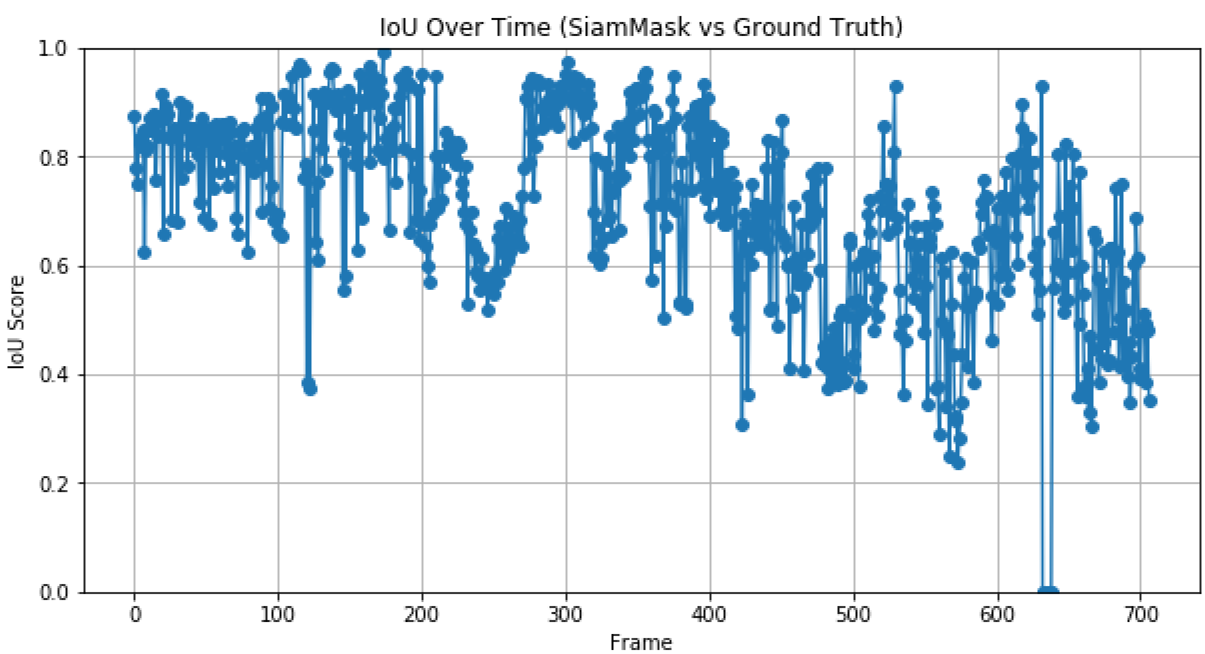

Figure 7 IoU vs. Frame with helicopter sequence using SiamMask

Comparison between Kalman filter and SiamMask

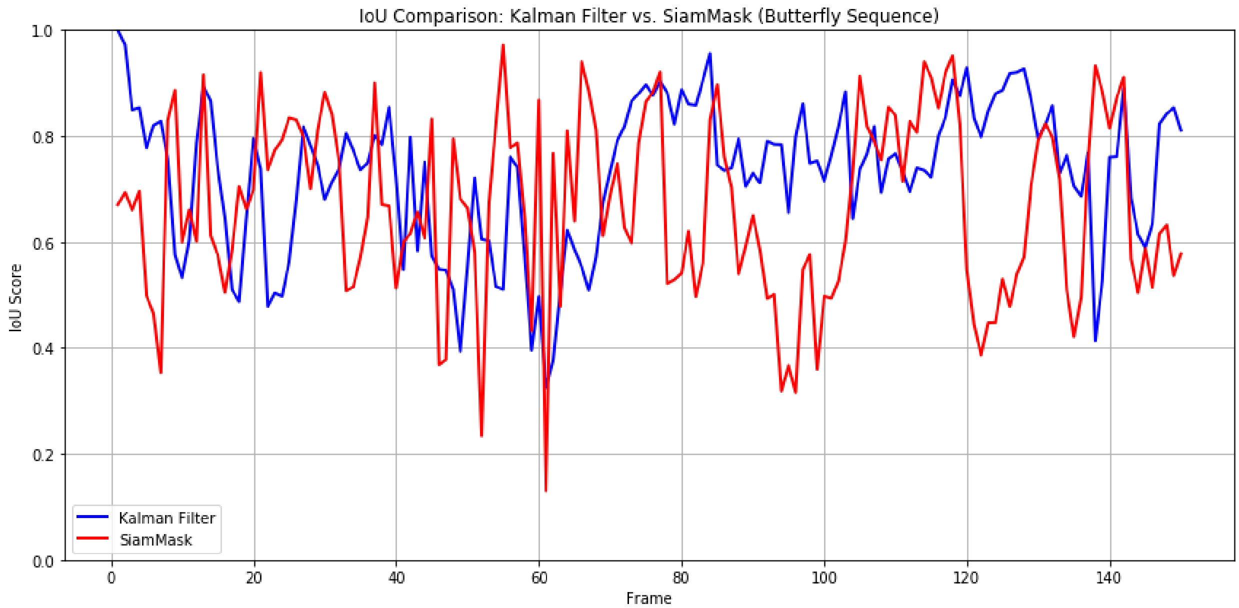

Figure 8 SiamMask vs. Kalman Filter (Butterfly)

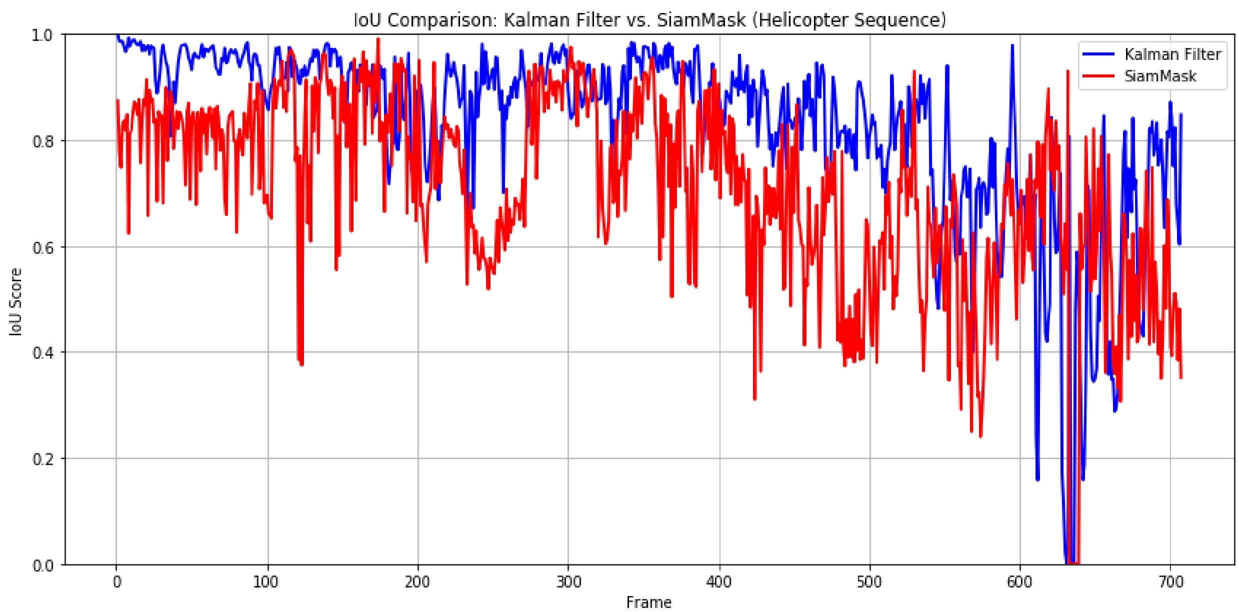

Figure 9 SiamMask vs. Kalman Filter (Helicopter)

Sample frames

Figure 10 Kalman Filter Prediction in Butterfly Sequence

Figure 11 Kalman Filter Prediction in Helicopter Sequence

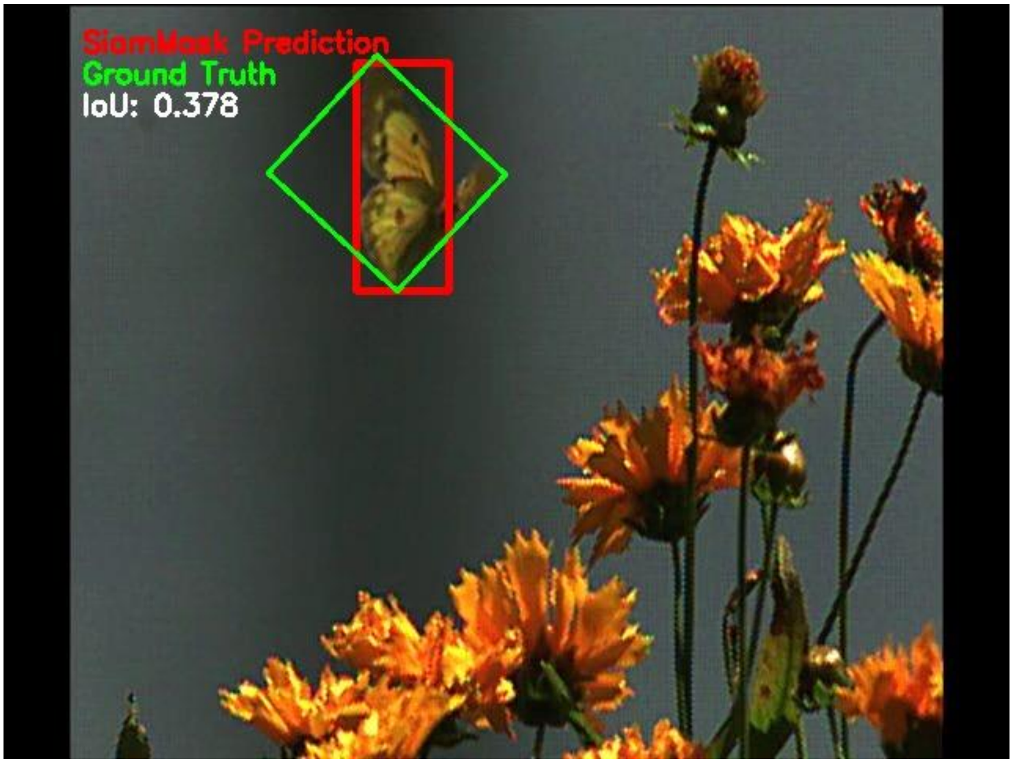

Figure 12 SiamMask Filter Prediction in Butterfly Sequence

Figure 13 SiamMask Filter Prediction in Helicopter Sequence